

## REMARKS

Applicant respectfully requests reconsideration of the present application in view of the foregoing amendments and in view of the reasons that follow.

Claims 1-8, 17, 20, and 21 are currently being amended.

This amendment adds, changes and/or deletes claims in this application. A detailed listing of all claims that are, or were, in the application, irrespective of whether the claim(s) remain under examination in the application, is presented, with an appropriate defined status identifier.

After amending the claims as set forth above, claims 1-23 are now pending in this application.

In the outstanding Non-final Office Action of January 5, 2009, the Examiner rejected claims 20 and 21 under 35 U.S.C. § 112, second paragraph because the recited limitations “the client device” and “the consumer user interface” lack antecedent basis. In response to the Examiner’s rejection, Applicant has amended claim 20 to more particularly recite “the system is a mobile telephone.” Additionally, Applicant has amended claim 21 to more particularly recite “a consumer user interface.”

Claims 1-19 were rejected under 35 U.S.C. § 101 because in the Examiner’s opinion, claims 1-19 are directed to non-statutory subject matter. In response to the Examiner’s rejection, claims 1-7 have been amended to more particularly recite a “computer-implemented method” in accordance with the Examiner’s suggestion at page 6 of the outstanding Office Action. Claim 8 has been amended to more particularly recite that the claimed device comprises a processor and a memory unit operatively connected to the processor, including a consumer application, at least one provider application, and an application interworking framework. Claim 17 has also been amended to more particularly recite that the claimed consumer application, provider application, and application interworking framework are each computer-implemented.

Claims 1-3 and 5-23 were rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by U.S. Patent No. 7,194,743 (Hayton et al.) Applicant traverses the rejection for the reasons set forth below.

It should be noted that Applicant incorporates herein by reference in their entirety, the arguments regarding the rejection of claims 1-3 and 5-23 as presented in Applicant's April 24, 2008 and October 20, 2008 Replies.

With regard to independent claims 1, 8, and 17 of the present application, the Examiner has maintained his assertion that Hayton et al. teaches all of the required limitations recited therein. At page 3 of the previously issued Office Action of July 18, 2008, the Examiner asserted that one of ordinary skill in the art would recognize that UI elements are features/components/properties/content as allegedly evidenced by the providing of "features or functionalities to the UI application 42 such as 'Albert the Boss', 'Bert the manager', 'Cathy the underling', 'Current Salary', 'Employee', etc."

In particular, Applicant again submits that Hayton et al. fails to teach or suggest providing a feature to a consumer application, requesting a feature matching a consumer interest from an application interworking framework, and/or a provider application where an interface is provided for the provider application and a consumer application such that a feature interest is matched with a feature from the provider application. Moreover, the Examiner's reliance on examples of content/objects that would be displayed in an application is misplaced.

At pages 6-7 of the outstanding Office Action, the Examiner maintained that the client UI 42 described in Hayton et al. is analogous to the claimed consumer application and that Applicant's arguments were already addressed. Applicant submits that at pages 4-5 of Applicant's April 24, 2008 Reply, Applicant set forth various reasons why the Examiner's interpretation of Hayton et al. were/are incorrect. In the July 18, 2008 Office Action, the Examiner maintained his reasoning and put forth an argument allegedly supporting his position (the above-described evidence comprising "Albert the Boss", "Bert the manager", etc.) Thus, in Applicant's October 20, 2008 Reply, Applicant repeated the previous arguments in conjunction with arguments to rebut this new alleged "evidence" set forth by

the Examiner. Therefore and contrary to the Examiner's position that Applicant merely repeated arguments and ignored that Examiner, Applicant submits that the arguments of the October 20, 2008 Reply simply incorporated Applicant's previous arguments from the April 24, 2008 Reply while rebutting Examiner's additional "evidence." (*See, e.g.*, page 9 of the October 20, 2008 Reply).

At pages 7-8 of the outstanding Office Action, the Examiner further asserted that Hayton et al. is directed to "dynamically add new elements to the UI application" as allegedly evidenced by Column 19, lines 19-29 of Hayton et al. which describes a static and dynamic aspect of UI 42. Applicant disagrees with the Examiner's interpretation of Hayton et al.

As previously indicated by Applicant, Hayton et al. is directed to a system and method of providing, e.g., remote access to an application. That is, Hayton et al. teaches providing a user-interface (UI) portion of an application to either the same machine on which the application is executing, or on another machine remote from the machine executing the application. (*See, e.g.*, Abstract and Column 2, lines 44-52). A user that is, e.g., customizing a UI 42, such as a web page, may choose UI elements 46 and associate those chosen UI elements 46 with one or more properties of an application component 34 by indicating one or more property paths. (*See, e.g.*, Figures 1-4 and Column 11, lines 3-15). Additionally, Hayton et al. suggests that an application 26 (which includes, e.g., the application components 34), can create or delete properties. (*See, e.g.*, Column 12, lines 44-50). Hence, Hayton et al. is merely directed to a single application (e.g., the server application 26) that has a client UI 42 which can be "customized" with various UI elements 46. Moreover, Applicant submits that Hayton et al. is clearly directed to a "development" environment, where users can, e.g., "develop" the appearance of the UI, e.g., a web page or employee salary application. (*See, e.g.*, Figure 4 and 5, Column 9, lines 19-29, and Column 10, lines 55-65).

With regard to the static and dynamic aspect of the UI described in Hayton et al., Applicant submits that Hayton et al. is clear in indicating that a developer may develop the static aspect of the UI, such as a UI "template." The UI "form" or "template" is "not modified" during execution, but only the values that appear in the fields of the form."

(emphasis added). (*See, e.g.*, Column 9, lines 7-9 of Hayton et al.) The dynamic aspect of the UI 42 of Hayton et al. on the other hand is clearly described as consisting of “changes to the static page ‘template’ such as filling in fields, changing text values and a navigation through a number of dialogs or pages under the control of the user and/or the application.” (emphasis added). (*See, e.g.*, Column 9, lines 5-13 of Hayton et al.) Hayton et al. goes on to further explicitly state that “[t]he invention adds only the ability to fill in a page template dynamically” (emphasis added) at, e.g., Column 9, lines 17-28.

Therefore, Applicant again submits that any changes to an application or to the UI 42 of Hayton et al. are merely the result of a developer utilizing, e.g., some type of HTML editor, as described at, e.g., Column 15, lines 11-15. Moreover, any “dynamic” changes to the UI 42 do not amount to adding/providing a new feature to the UI 42, but rather merely dynamically providing values/data to populate the UI 42 via property paths (as well be discussed below in detail).

In contrast to the above-teachings of Hayton et al., Applicant again submits that various embodiments disclosed in independent claims 1, 18, and 17 of the present application, require adding features to a consumer application, where the feature matches that which a consumer wishes to have. As described above, Hayton et al. at best, merely teaches adding a UI element to the UI 42, not a feature to application 26. (*See, e.g.*, Figure 4 and Column 20, lines 21-64). See also Column 15, lines 9-20 of Hayton et al. where it is described that a “user” creating a page via, e.g., an HTML editor may locate UI elements using the page interface 112 (which Hayton et al. considers to be development software, such as the HTML editor) from a set of predefined elements and associates that element with a property path which ultimately directs the element to a value or other data for populating the element.

Additionally, Applicant previously asserted at, e.g., page 10 of the October 20, 2008 Reply, that the UI 42 of Hayton et al. cannot be reasonably interpreted to read on the claimed consumer application because the UI 42 of Hayton et al. as is supported by its description, a front-end to an application running on a server. At page 8 of the outstanding Office Action, the Examiner asserted that “UI 42 is considered as consumer application. Again, Examiner

already addressed this argument in the previous action.” Applicant emphatically disagrees with the Examiner in that the Examiner has failed to properly rebut and/or substantially answer Applicant’s arguments. That is, in the October 20, 2008 Reply, Applicant as described above, set forth explicit arguments as to why the UI 42 of Hayton et al. could not be considered to read on the claimed consumer application (as first asserted by the Examiner in the July 18, 2008 Office Action). Hence, Applicant rebutted the Examiner’s assertions with reasoned arguments. It is improper now for the Examiner to merely restate his position without any evidence to further support his assertions and/or rebut Applicant’s arguments.

Further with regard to, e.g., claim 1, the feature matching a consumer interest is requested from an application interworking framework. It appears from the Examiner’s assertions that the Examiner is reading, e.g., the property connector API 22 as the claimed application interworking framework. (*See, e.g.*, Column 11, lines 49-52 of Hayton et al. where it is described that upon execution of the property connector API 22, a UI element 46 is mapped to an application component 34). However, nowhere in the portions of Hayton et al. cited by the Examiner, nor anywhere else in Hayton et al., is it taught or even contemplated that any request is made of the property connector API 22. Moreover, claim 1 further requires, e.g., identifying a provider and providing a feature if the provider is identified. Applicant submits that nowhere in Hayton et al. is it described or otherwise suggested that a provider is identified. As noted above, Hayton et al. is directed to modifying, e.g., a UI 42, where the same server process 14/application 26 is always associated with the UI 42. Hence, there is no need for the system and method of Hayton et al. to ever “identify a provider” as required by independent claim 1 of the present application.

At pages 8-9 of the outstanding Office Action, the Examiner responded to Applicant’s above-described argument and asserted that:

Upon execution of the property connector API 22, a request is initiated to download the UI page 42 that containing the UI elements 46. A [sic] details of the server node 60 connects to the computing device is received. The details of the server indicated what server the computing device is connected to. Even assume that Hayton fails teach “provider is identified.” It would have been obvious to one having ordinary skill in the art at the time the invention was made to use the details of the

server node 60 received by the computing device to identify the server it connected to for security purposes.

Applicant submits that nothing in Hayton et al. nor in the Examiner's above assertion supports the position that Hayton et al. teaches "requesting from an application interworking framework a feature matching a consumer interest of a consumer application." That is, Hayton et al. merely describes that a request is made to deliver "the page [UI] 42." Applicant is at a loss as to how requesting the downloading of the UI 42 is in any way analogous to requesting "a feature matching a consumer interest of a consumer application." Even if, the UI 42 elements (as alleged by the Examiner) are likened to a feature, the API 22 of Hayton et al. does not request any elements, but the entire UI.

In fact, Applicant directs the Examiner to, e.g., Column 8, lines 51-57 of Hayton et al. that describe that "a third party could generate a user interface for a 'published' application without any access to the application code or runtime environment... A third party could design a new client type without the server's involvement." Additionally, Column 11, lines 26-30 of Hayton et al. explicitly states that, "[t]he property connector API 22, and thus the client portion 22a and the server portion 22b, is a process that is independent of the application 26 (i.e., not a part of nor generated from the application 26)." Hence, the Examiner's reliance on the alleged obviousness and/or inherent connection or involvement of the server is unsupported, as is any execution or requesting in association with the API. That is, Hayton et al. is clear in that only elements of UI, not the application to which the UI is acting as the front-end/interface is being developed/created. Applicant is unaware of how a feature may be added to an application without involving the application. Again, Hayton et al. is clearly directed merely to creating a user interface, not an application, and then assigning property paths linking the user interface elements to data/information/etc. dynamically.

Furthermore, Applicant notes that the Examiner has rejected claims 1-3 and 5-23 under 35 U.S.C. § 102(b) but has presented arguments relying on what is allegedly obvious to one of ordinary skill in the art, which would necessitate a rejection under 35 U.S.C. § 103(a), not 102(b).

At pages 9-10 of the outstanding Office Action, the Examiner further asserted that “Hayton merely teaches dynamically adding UI elements to the UI application not dynamically adding data to the UI elements.” These assertions were in response to Applicant’s previously presented arguments that Hayton et al. merely teaches that a developer may design a UI 42, using, e.g., an HTML editor, as already described above, and linking UI elements to various values, text, etc. via the use of property paths. Applicant maintains that the Examiner’s interpretation of Hayton et al. is incorrect.

Applicant again directs the Examiner to, e.g., Column 9, lines 17-18 of Hayton et al., where it is explicitly stated that “[t]he invention adds only the ability to fill in a page template dynamically.” (emphasis added). As described at, e.g., Column 10, line 55-Column 11, line 48 and Column 15, lines 6-19 of Hayton et al., a user, i.e., a developer using an HTML editor, for example, creates/modifies a UI by selecting or adding UI elements to the UI and linking (using a property path) a UI element to data/types of data associated with an application component so that information can be used to dynamically populate the UI elements. For example and as described at, e.g., Column 12, lines 40-59 of Hayton et al., an application (not an end-user) may create/delete application components (not UI elements) based on such property paths that provide a loose connection to, e.g., data, based on data type. Thus, if a particular property path changes or no longer exists, this will not affect the UI. However, in no way is this analogous to any of the limitations recited in independent claims 1, 8, and 17 of the present application.

With regard to independent claim 21 of the present application, Applicant previously presented arguments indicating that claim 21 requires storing a user interface element corresponding to the consumer application interest is required. As described above, Hayton et al. does not teach any sort of consumer application interest, only a UI element that a user may be interested in utilizing in the UI 42. Furthermore, the section of Hayton et al. that the Examiner cited to support his position, e.g., Column 16, lines 31-32, does not in any way anticipate this claimed feature of the present application. That is, Column 16, lines 31-32 merely indicate that values corresponding to application components 34 are stored. In contrast, claim 21 of the present application requires that a user interface element is stored.

Furthermore, and in contrast to the Examiner's continued assertion that Hayton et al. teaches a consumer application (i.e., UI 42 of the client) and a provider application (i.e., application 26 of the server), Applicant submits that the Examiner is mischaracterizing the system and method of Hayton et al. Figure 1 of Hayton et al. illustrates "an embodiment of a system 10 for communicating changes between a user interface and an executing application, using property paths." (*See, e.g.*, Column 9, lines 65-67 of Hayton et al.). That is, Figure 1 of Hayton et al. is indicative of a UI of an application that may be running remotely from the server providing the application, where any changes to, e.g., the content/data populating different fields/boxes/aspects of the UI can be determined by property paths. In no way does Hayton et al. suggest the existence of two applications, a consumer application and a provider application. The Examiner's interpretation that a UI reads on the claimed consumer application is simply untenable. Nothing in Hayton et al. suggests this interpretation. Rather and again, the UI of Hayton et al. is merely the UI associated with the application hosted by the server. Moreover, Applicant submits that the use of language distinguishing a "UI" from an "application" clearly indicates that Hayton et al. does not consider the UI to be an actual application, let alone a consumer application.

Moreover, claim 21 requires communicating the user interface element to an application interworking framework. However, Hayton et al. does not teach or even contemplate such an operation. Hayton et al. merely teaches that the "user interface portion of the application can be delivered to the computer user..." and that the server portion 22b transmits to the client portion 22a any change..." As described above, a user of Hayton et al. chooses a UI element and, e.g., associates that UI element with a state of property of an application component. However, communicating that UI element to the property connector API 22 is never taught or suggested. Second, as described above, it appears that the Examiner has interpreted the property connector API 22 of Hayton et al. as allegedly reading on the claimed application interworking framework. Given this interpretation, the operation involving "the server portion 22b transmits to the client portion 22a" would be analogous to the property connector API 22 communicating with itself because the server portion 22b and client portion 22a are both a part of the property connector API 22, and hence do nothing to support the Examiner's assertions. (*See, e.g.*, Column 11, lines 23-30 of Hayton et al.)



In response to the Examiner's assertions at pages 10-12 of the outstanding Office Action, that Hayton et al. allegedly teaches 2 applications (in the Examiner's opinion, the UI 42 and the application 29) and that UI elements are "transferred from the server to the client device," Applicant again submits that nothing in Hayton et al. supports these positions. In particular, and again, Hayton et al. is clear in the language used to describe the UI 42 and the application 29. That is, Hayton et al. does not consider the UI 42 to be an application, but rather a front-end interface for the application 29. Applicant is aware that the Examiner may interpret Applicant's own claims in the broadest reasonable manner, but to interpret an applied prior art reference in a manner not commensurate with the teachings therein is improper. The Examiner cannot simply contradict the explicit teachings of an applied reference and apply his own interpretation.

Applicant further submits that Column 11, lines 37-52 do not teach or suggest in any way that "UI elements are transferred from the server to the client device." To wit, this cited section of Hayton et al., as acknowledged by the Examiner merely states that "the property connector API 22 maps each dynamic user-interface element 46 to a property 38 of an application component 34 using the associated property path." As already discussed above, the mapping using the property path merely indicates a link from the UI element to a piece of data/data type. The fact that the UI element is "mapped" indicates that there is no actual transfer of the UI element – hence the action of mapping. As would be clearly understood by those of ordinary skill in the art, to map indicates linking/pointing to some data, not actually transferring, e.g., a UI element. In other words, a developer in Hayton et al. would have already created the UI 42 with the requisite elements. See for example, Column 25, lines 6-10 of Hayton et al., which states that:

The iterator type predefined UI element 78 includes a template and is linked to an indexed property path. The template represents the layout of the iterator type predefined UI element 78 for a single member (i.e., a single index value) of the property path.

Additionally and as to the Examiner's assertions at page 10, "[t]he server node must store the UI elements in order to provide these elements to the client node. If the UI elements are not stored at the server node, where can it be?" Applicant submits that the Examiner's

assumptions/questions are premised on his misinterpretation of Hayton et al. That is and again, UI elements are not transferred/sent anywhere during execution of the application. As described above, Hayton et al. merely teaches that a developer may develop, e.g., a static UI or template, with predefined elements included therein. The dynamic aspect to the UI 42 merely occurs via property paths defined by the developer which allows the UI elements to be “filled in” with, e.g., data/values/etc.

With regard to the Examiner’s remaining assertions at pages 12-14 of the outstanding Office Action, Applicant submits that the arguments set forth above (in conjunction with Applicant’s previously presented arguments that are incorporated by reference herein) rebut/address the Examiner’s position/reasoning with respect to independent claims 1, 8, 17, and 21 of the present application. Thus and again, Applicant submits that Hayton et al. fails to teach each and every limitation recited in independent claims 1, 8, 17, and 21 of the present application.

Applicant again further submits that the Examiner has mischaracterized the teachings of Hayton et al. with respect to dependent claims 2, 3, 5-7, 9-16, 18-20, 22, and 23 of the present application. For example and with respect to, e.g., dependent claims 2, 12, and 18, the Examiner asserted that the claimed limitation of “using generic parameter in application interworking framework application programming interfaces (API)” is taught by Hayton et al. at Figure 1 and Column 11, lines 50-52.

The Examiner also had previously asserted that the associated property path taught by Hayton et al. could be “interpreted” to be a generic parameter and thus, allegedly teaching mapping each dynamic user interface. Again, Applicant submits that the property path of Hayton et al. is utilized for directing an application as to where/how to receive, e.g., object data or values, that are to populate a web page UI, for example, and in no way teaches or suggests the use of generic parameters. As already described above, these property paths are paths to particular information/values/data, and therefore cannot merely be generic parameters. Even if the Examiner’s intent was to suggest that the “syntax” of a property path was somehow generic and that the actual property path itself could be dynamically changed,

again, the property path is applicable merely to data/values that would populate a UI element, such as a text box.

Further still and in the outstanding Office Action at page 14, the Examiner asserted that he has interpreted the term “generic parameter” as any parameters used by the API of Hayton et al. (e.g., property path, UI elements, etc.) Applicant disagrees. First, Applicant submits that the term “generic” is well known and understood by those of ordinary skill in this and other arts. Moreover, paragraphs [0009]-[0016] and [0049]-[0060] describe and provide examples of “generic” parameters. Thus and contrary to the Examiner’s assertions, generic parameter as claimed and in light of the specification of the present application is sufficiently clear, not to mention a term that is well understood.

Second, Applicant submits that the Examiner is contradicting himself by asserting (as described above) that at least UI elements can be interpreted as the claimed generic parameter. As is clear from the Examiner’s position, UI elements of Hayton et al. (although incorrectly) are already being interpreted as the claimed feature being added to a consumer application. Applicant submits that the Examiner cannot now assert that the UI elements of Hayton et al. also read on an entirely different claim element, or at least without further explanation.

With regard in particular to, e.g., claim 11, the Examiner previously asserted that the claimed limitation “wherein the new consumer application integrates into the device as if part of an original group of software applications for the device” is allegedly read on by Hayton et al. at Column 10, lines 66-67. Column 10, lines 66-67 of Hayton et al., as quoted by the Examiner merely states that “[t]he client process 18 produces a user-interface (‘UI’) 42 that is displayed to a user.” Applicant submits that producing a UI that is displayed to a user suggests nothing even remotely associated with integrating a new consumer application as if part of an original group of software applications as required by, e.g., claim 11 of the present application. First, and as described above, Hayton et al. fails to teach the integration of any new consumer application. Rather, Hayton et al. is directed to implementing UI elements in a UI 42. Second, the fact that a UI 42 is displayed indicates nothing regarding how a new application is integrated into an original group of applications as if it were an original part

thereof. There is simply no language or implicit indication in Hayton et al. that suggests such a feature.

In the outstanding Office Action at page 15, the Examiner responded to Applicant's previous arguments regarding claim 11 by indicating that he is interpreting claim 11 "to further clarify that the software application is integrated into the device." Applicant submits that regardless of how the Examiner wishes to interpret the claim, he is completely ignoring the language of claim 11 indicating that the new consumer application integrates into the device "as if part of an original group of software applications for the device. Applicant yet again (as indicated above) directs the Examiner to paragraphs [0003]-[0004] of the present application where it is described that certain prior art that the present application seeks to improve upon includes, e.g., devices such as phones. When new features/applications are added to such devices, the new features/applications do not look/act like those features/applications originally provided by the phone manufacturer.

Additionally, Applicant submits that because dependent claims 2, 3, 5-7, 9-16, 18-20, 22, and 23 are dependent upon independent claims 1, 8, 17, and 21 of the present application, Hayton et al. fails to teach all of the required limitations recited in the dependent claims for at least the same reasons as discussed above with regard to, e.g., claims 1, 8, 17, and 21.

Claim 4 was again rejected under 35 U.S.C. § 103(a) as being unpatentable over Hayton et al. in view of International Patent Application No. WO 00/58855 (Gudmunson).

With regard to claim 4, the Examiner correctly recognized that Hayton et al. does not teach the use of dynamic link libraries. However, the Examiner asserted that Gudmunson cures this deficiency. Applicant again submits that because Gudmunson was applied by the Examiner solely for purpose of evidencing the use of DLLs, Gudmunson cannot cure the deficiencies of Hayton et al. described above. Therefore, because claim 4 depends from independent claim 1 of the present application, Applicant submits that the alleged combination of Hayton et al. and Gudmunson still fail to teach all of the required limitations of claim 4 for at least the same reasons as discussed above.

Because none of the references cited by the Examiner, either separately or in combination with each other, teach all of the required limitations of independent claims 1, 8, 17, and 21 of the present application, Applicant submits that each of these independent claims are patentable over this prior art. Furthermore, because dependent claims 2-7, 9-16, 18-20, 22, and 23 are each directly or indirectly dependent upon independent claims 1, 8, 17, and 21, Applicant submits that each of these claims are allowable for at least the same reasons as discussed above.

Applicant believes that the present application is now in condition for allowance. Favorable reconsideration of the application as amended is respectfully requested.

The Examiner is invited to contact the undersigned by telephone if it is felt that a telephone interview would advance the prosecution of the present application.

The Commissioner is hereby authorized to charge any additional fees which may be required regarding this application under 37 C.F.R. §§ 1.16-1.17, or credit any overpayment, to Deposit Account No. 19-0741. Should no proper payment be enclosed herewith, as by a check being in the wrong amount, unsigned, post-dated, otherwise improper or informal or even entirely missing or a credit card payment form being unsigned, providing incorrect information resulting in a rejected credit card transaction, or even entirely missing, the Commissioner is authorized to charge the unpaid amount to Deposit Account No. 19-0741. If any extensions of time are needed for timely acceptance of papers submitted herewith, Applicant hereby petitions for such extension under 37 C.F.R. §1.136 and authorizes payment of any such extensions fees to Deposit Account No. 19-0741.

Respectfully submitted,

Date 5 May 2009

By /G. Peter Albert, Jr./

FOLEY & LARDNER LLP  
Customer Number: 30542  
Telephone: (858) 847-6735  
Facsimile: (858) 792-6773

G. Peter Albert Jr.  
Attorney for Applicant  
Registration No. 37,268